DOCUMENT RESUME

ED 147 274 95 SP 011 851

AUTHOR Coulter, Cynthia L.
TITLE Training Observers for Naturalistic Observational Research. Report No. 76-9.
INSTITUTION Texas Univ., Austin. Research and Development Center for Teacher Education.
SPONS AGENCY National Inst. of Education (DHEW), Washington, D.C.
PUB DATE Jun 76
CONTRACT NIE-C-74-0089; OEC-6-10-108
NOTE 24p.

EDRS PRICE MF-$0.83 HC-$1.67 Plus Postage.
DESCRIPTORS *Classroom Observation Techniques; *Data Collection; Data Processing Occupations; Educational Research; *Project Training Methods; *Training Techniques

ABSTRACT

Procedures are presented for training classroom observers to collect data in naturalistic research projects. Training methods are directed toward the collection of highly reliable observational data. Pre-classroom training prepares observers for data collection by working with the coding manual, video-tapes, and written dialogues, and through group discussions with the trainer. A list of practical guidelines for coders to follow while collecting data helps to minimize observer effects, maintain good relations with school personnel, and insure high quality data. Techniques for training in the classroom are described, in addition to ways of dealing with difficulties that may arise in using the observational system. A discussion of inter-coder reliability describes methods of obtaining reliability estimates and also analyzes the problems that may arise when attempting to establish a high degree of reliability among coders. The training procedures presented in this report can be used in either small- or large-scale research projects and are appropriate for training observers to use both low- and high-inference observational systems. (Author)

****
* Documents acquired by ERIC include many informal unpublished *
* materials not available from other sources. ERIC makes every effort *
* to obtain the best copy available. Nevertheless, items of marginal *
* reproducibility are often encountered and this affects the quality *
* of the microfiche and hardcopy reproductions ERIC makes available *
* via the ERIC Document Reproduction Service (EDRS). EDRS is not *
* responsible for the quality of the original document. Reproductions *
* supplied by EDRS are the best that can be made from the original. *
****

ED147274

SCOPE OF INTEREST NOTICE
The ERIC Facility has assigned this document for processing to: SP TM
In our judgement, this document is also of interest to the clearinghouses noted to the right. Indexing should reflect their special points of view.

Training Observers for Naturalistic Observational Research

Cynthia L. Coulter

U.S. DEPARTMENT OF HEALTH, EDUCATION & WELFARE
NATIONAL INSTITUTE OF EDUCATION
THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT. POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL NATIONAL INSTITUTE OF EDUCATION POSITION OR POLICY

Report No. 76-9
Research and Development Center for Teacher Education
The University of Texas at Austin
June, 1976

SP011 851

This project was supported by the National Institute of Education Contract OEC 6-10-108, Research and Development Center for Teacher Education, and by Contract NIE-C-74-0089, Correlates of Effective Teaching. The opinions expressed herein do not necessarily reflect the position or policy of the National Institute of Education, and no official endorsement by that office should be inferred.

This report will outline methods of training classroom observers to use complex data collection instruments. The goal of these training procedures is for the trainees to achieve a high degree of intercoder reliability in using observational systems.

The methods presented here have been used to train large groups of coders to gather reliable field data in large-scale research projects, using low inference classroom observational systems. However, the methods can be adapted easily to smaller projects using fewer coders, and are appropriate for use in training with high inference data collection instruments as well.

Although research projects will differ in size, types of data to be collected, goals, etc., the training procedures discussed here are necessary in almost any project. They deal with the specifics of day-to-day problems and with the practical considerations that must be dealt with in every research endeavor.

Coder training is one of the most critical processes involved in conducting a research project. No matter how sophisticated the observational system, valid and reliable data cannot be obtained unless the coders use the system correctly. Therefore, as much thought and effort should be put into thorough training of coders as was put into designing the system they will use. The coding trainer must be expert in using the system and must also be able to teach others effectively.

Training can be divided into 2 major stages: (1) training prior to going into the classroom, during which coders will familiarize themselves with the system and learn how it is used, and (2) practice <u>in</u> the classroom, aimed at obtaining acceptable inter-coder reliability estimates.

Pre-Classroom Training

Although it may seem desirable to send coders into the classroom right away, we have found that a period of orientation and training before going into the classroom minimizes coder difficulty and actually saves time. Coders often find themselves confused and frustrated the first time they attempt to use an observational system; events in classrooms move at such a fast pace that coders can't begin to keep up with their coding. Also, the noise level often is quite high. This imposes enough distraction that coders may have difficulty in hearing what is said and in learning to code efficiently. In addition, anything the coder is unclear about immediately becomes evident; this may prevent coders from getting much worthwhile practice, because they are not sure if they are using the system correctly. Therefore, we recommend that coders be as familiar as possible with the system before attempting to code in actual classroom settings.

Pre-classroom training can and should be broken down into 3 or more steps: (1) orienting the coders to the research project, the observational system, and the types of data to be collected, (2) explanations (both verbal and written) of the categories in the system and how they are used, and (3) simulated practice in using the observational system.

Orientation

The first step in coder training should be an orientation session, in which the study should be outlined (if there are no hypotheses which require "blind" observers) and the coders given some idea of what their job will be.

For those unfamiliar with observational research, a general discussion of research techniques will be needed. Looking at classroom interactions in terms of discrete variables may be a new concept for many, and this will need to be explained. Coders may also need to understand the difference between low and high inference data. They need to know whether they are expected to be rather passive recorders of low inference information, or be more judgmental and evaluative as in the case of high inference data collection.

Probably the best way to give coders a feel for their job would be to discuss the research questions or issues that are to be investigated, and show how the observational system was designed to answer these questions. This familiarizes them with the study as a whole and gives them some idea of what will be expected of them. Any observational system can be subject to coder bias, so the project director will need to consider the effects on the data collected if the coders are aware of the hypotheses involved in the study. If the design of the project necessitates "blind" coders, this discussion would be phrased in more general terms, using hypothetical situations. In any case, coders need to understand the basis for the observational system, and where their job fits into the total picture.

After this initial orientation, the observational system should be presented. Coders should have copies of the system to look at as it is discussed. The system should be described in its larger sections first, and then each individual category should be explained fully. This discussion should also deal with how the system is actually used, especially if sequential events are to be recorded or if coding is to take place only during certain times during the class period.

Learning to use an observational system can be a difficult task for many people, and a good deal of confusion and frustration is almost unavoidable during certain stages of training. If the observational system is fairly complex, trainees are apt to become very apprehensive over what they are expected to learn. The trainer should try to reassure the coders, letting them know that they will be able to learn the system, and that things will become more clear as they go along.

The Coding Manual

A functional coding manual is an absolute necessity in coder training. This manual should contain a thorough discussion of the coding system, explanations and descriptions for each category with clear, relevant examples, and instructions for how and when the system is used. Each coder should have his/her own copy of the coding manual during the training period and also during data collection for use as a reference and "refresher."

The coding manual should be kept strictly up to date. This is important not only for the coders' sake, but also for future data analysis. If the coding of a particular category is changed, this may well affect later data reduction and analysis; and if no one can remember just how the category was coded, the meaning of the data will be lost.

Following the explanation of the observational system, coders will need a period of time to read and study the coding manual. For most systems, a cursory reading of the manual will not be sufficient. Every effort should be made to see that coders spend a good deal of time going over the manual, perhaps even to the point of "overlearning" the system. In our

research, we have found that coders very nearly had to memorize the coding manual before highly reliable and valid data could be collected. As they study the coding manual, coders should write down any questions they may have, so that these can be discussed. Some researchers may find it useful to administer a written test to insure that coders are familiar with the material in the manual, and to serve as feedback to the coders as to what areas of the system they may not fully understand.

After studying the coding manual, coders need an opportunity to ask questions and discuss the system with the trainer. Doing this as a group is most efficient, because the trainer will not have to repeat explanations and answer the same questions for every individual, and because groups will raise more questions than individuals will. Thinking about these questions and the answers to them facilitates understanding of the coding system and how it is used.

After reading the manual, a useful exercise for coders is to have them write out brief examples of each variable in the observational system. In doing this, coders quickly find out which variables or sections of the system they do not understand and can go back and re-read parts of the manual.

Coding Dialogues

At this point, coders should have a general understanding of the observational system. The next step in training involves having trainees "code" written transcripts of teacher-student interactions. These dialogues should be prepared ahead of time by the trainer, and should include descriptions of appropriate events or interactions. An effort should be made to

include a number of examples for each category in the system, so that coders get experience with all types of interactions. It is best to provide each coder with a series of coding dialogues graduated in difficulty. The first (and easiest) dialogues should involve only the simplest and most obvious discriminations. After coders have mastered these, they can go on to more difficult dialogues which include more ambiguous and hard-to-code interactions. The trainees themselves can also make up practice dialogues (and keys to go with them) and code one another's dialogues for extra practice.

In coding the practice dialogues, coders should use the same coding sheets they will be using in the classroom. Also, coders should be free to refer to their coding manuals as they work with the dialogues. This practice helps coders learn to use the system, teaches them to discriminate between categories, and familiarizes them with the mechanics of exactly where categories are located on the page. (This becomes very important when coders go into the classroom, because they often will not have time to scan the page in order to figure out where to put a checkmark).

Coders can work independently or in pairs or groups on the practice dialogues, but then they should get together as a group to check what they have done and discuss mistakes. When coders make mistakes, it is important that they understand why the coding is incorrect, and a good deal of discussion may be necessary before coders understand certain discriminations. For this reason, we recommend group discussions in checking coding dialogues rather than simply providing a key showing the correct coding.

Coding of practice dialogues familiarizes coders with the use of the observational system, and discussions of the correct coding for each dia-

logue can help them to really understand the system. Practice dialogues also can be used as evaluative tools for assessing how well coders have learned the system and as the basis for giving feedback to coders on how they are progressing. For this, coders should be required to code dialogues independently, without referring to the coding manual or one another. Their coding then can be "graded" by the trainer and returned. Special attention should be given to any interactions in the dialogue that were miscoded frequently, because these indicate problem areas for the coders (and perhaps the coding system itself). Discussing these with the group may be sufficient to clear up misunderstandings. However, it also is advisable to add more specific (even redundant) material to the coding manual in the sections with which coders are having difficulty.

Questions from Coders

At this point in training, coders typically begin exercising their creative abilities in thinking up "what if" questions: "What if the teacher asks a question, and the whole class answers? How do we code that?" Some of these "what if" questions stem from an incomplete knowledge of the observational system, so that many of these questions will be covered in the coding manual. However, other questions may involve situations not dealt with in the coding manual. In most cases, the manual will not have been specific enough in describing how the system should be used. In these instances, it will be necessary to insert additional material into the coding manual for specifying how these events should be coded.

Other "what if" questions may reveal a deficiency in the observational system, and trainers should be prepared to acknowledge and respond to this

problem if it occurs. For instance, if your research is concerned with student response to teacher questions, your system likely will have categories for right answers, wrong answers, and "don't knows." It won't take long for coders to start asking "What if the teacher asks a question and the student doesn't respond at all?" If your system cannot categorize this situation, you may want to add a "no response" coding category to the observational system, and rewrite or add to the relevant sections in the coding manual. Another alternative would be to decide to code "no response" under "don't know" category. If this is the case, the coding manual should specify that "no responses" are to be coded in this way. Establishing a coding "convention" such as this often becomes necessary during coder training.

Some coders obviously delight in asking "what if" questions in the hopes of finding a "hole" in either the manual or the observational system. These questions must be dealt with as they arise and should not be discouraged. They indicate that coders are thinking about the coding system and the events that might occur in a classroom. However, questions asked by coders probably will be one of three main types: (1) questions asked by coders who don't thoroughly understand the observational system yet; (2) questions which indicate that the coding manual is not complete or specific enough in defining categories and describing how they are used; and (3) questions that point up a deficiency in the observational system itself, and which necessitate an alteration or addition to the system.

Use of Video Tape

The coding of video tapes may seem to be an ideal way to train coders. However, this method has several drawbacks that should be considered before

investing either money or training time in video-tapes. The primary disadvantage of video tapes is that coders often can't see what is happening well enough to code accurately. This is particularly a problem with split-screen tapes, in which one half of the screen is focused on the teacher and the other half shows the students in the class. On many tapes, it is impossible to tell which student the teacher is speaking to. This is a problem if you are trying to code who initiated the interaction (teacher or student), whether the student volunteered, and whether the teacher is maintaining a series of interactions with the same student vs. moving from one student to another. In addition, if the audio portions of the tapes are of poor quality, the sound of chairs scraping, feet shuffling, etc., may drown out the teacher's voice. Coders rapidly become frustrated and confused when a tape does not provide all the visual or auditory information that is needed to code accurately and completely.

The other major disadvantage of video tapes is that it may be difficult to find tapes that include classroom activities or interactions which are appropriate to your observational system. Grade level, subject matter, and classroom format are all important considerations in selecting video tapes to be coded with a particular observational system. Especially if the system is to be used only in certain classroom contexts or is focused on specific types of teacher or student behavior, it might be difficult to find appropriate tapes for training coders.

This is not to say that video tapes should never be used in coder training. It is possible for a useful video tape to be compiled by splicing together appropriate segments from a number of different tapes. A key for

the correct coding of this tape could be prepared in advance and used to score the accuracy of coding.

Another way in which video tapes can be useful is for coders to view them with the aim of coding only a few particular variables. As mentioned earlier, "complete" coding may be impossible. However, it is possible for trainees to code those relevant variables that can be picked up from the tape, and (for the time being) ignore the other categories in the system. For instance, if student responses and teacher feedback are clearly audible, coders could code using only these sections of the system and leave out the sections dealing with how students are selected for responding -- if this is not evident in the tape.

There also are some distinct advantages in using video tapes. One of these is that the tape can be stopped at any point; questions then can be asked, and the segment can be re-run and discussed. This is of course impossible in a live classroom. Viewing video tapes also gives the coders some idea of the typically fast pace of classroom events, and they can begin to build up some speed in their coding. Coders could even use video-tapes to obtain preliminary reliability estimates if the coding is not too difficult.

Another positive effect of viewing video tapes is that coders can better learn how to process or translate visual and auditory information into the categories on their coding sheets. They can begin to learn which pieces of information are irrelevant for their purposes, and which things to focus on and record. Experience in this cannot be gained from studying the coding manual or working with practice dialogues. Yet, being able to

process and interpret the barrage of stimuli present in a live classroom in the terms of the observational system is one of the abilities most critical to reliable data collection.

Finally, if a video tape can be obtained which is well suited for use with your observational system in terms of grade level, subject matter, and class activity, and which also provides clear and relevant examples (both visual and auditory) of the variables within your system, this tape could prove to be the most efficient and valuable training device.

Guidelines for Coders

Before coders are sent into classrooms for the first time, they need to be briefed on several important points concerning their behavior while in the schools. "Guidelines for Coders" should be prepared and given to each coder to explain and serve as a reminder concerning these points. We have found them important, not only for helping insure good data quality, but also for gaining and maintaining cooperation and good will from school personnel.

(1) Coders need to understand that the data they collect should be a representative sample of normal classroom functioning. Anything that interferes with classroom activity(like the presence of a coder) is apt to bias the data. Therefore, coders should try to minimize the effects of their presence in the classroom.

The most important thing for coders to remember is to remain as inconspicuous as possible. They should sit where they will not hamper or disrupt classroom activities. Coders should try to sit where they are able to see and hear what goes on in the room, but they also need to place them-

selves with discretion. Sitting up front by the teacher's desk may give the coder an excellent view of the teacher and the students, but this would be very awkward. The coder is likely to become the center of attention, and perhaps also be in the way of the teacher and students. It is preferable to sit at the sides or the back of the room at an empty desk or at a table.

Coders should remain seated, and not stand up or walk around the room. They also should avoid making the students or teachers feel self-conscious by craning their necks to see or by staring at them. Nothing is more apt to inhibit the behavior of the teachers and students than the feeling that they are being "watched." Of course, coders will not eat, smoke cigarettes, or engage in any other such inappropriate behavior while in the classroom. Anything that coders do which draws attention to themselves will disrupt the normal functioning of the class and should be avoided.

(2) Coders should be in the classroom before the bell rings or the class has officially begun. Arriving late and walking in during the class invariably disrupts the on-going activity to some extent. In addition, leaving the class early also should be avoided, not so much because it will bias data collection, but because it shows a lack of consideration for the teacher.

(3) Appropriate dress also must be considered. Here too, the goal should be to remain inconspicuous. Loud, sloppy, or revealing clothing not only draws attention to the coder, but may also offend some teachers or principals. Clothing should be clean, neat, and fairly conservative. Since dress standards vary widely, a good rule of thumb for coders is to dress like the teachers in the school.

(4) Another important point involves the coder's relationships with teachers. Coders should always be friendly and considerate toward teachers. They should introduce themselves to teachers the first time they observe in a class, and when they leave, they should thank them for their cooperation. When coders return many times to the same class, teachers get accustomed to their presence, so a smile or wave when entering or leaving is all that will be necessary to maintain a comfortable relationship. While remaining friendly, coders should also maintain a professional manner in dealing with teachers. An overly friendly relationship between the teacher and coder could cause a change in the teacher's normal behavior in the classroom. Also, coders should be careful never to talk to the teacher about other teachers or students. Attempts to get them to do so should be politely but firmly refused, with a reminder that all data collection and coder observations are confidential.

Teachers should be notified in advance for when they are to be observed. A weekly schedule given to each teacher is most convenient. Teachers should also be notified regarding changes in the schedule, especially if the coder is to observe at a time not previously scheduled.

(5) Coders should minimize personal interactions with students. They should not try to make friends with the students or engage them in conversation. Many students are curious as to why coders are there and what they are doing. Coders should always respond to direct student questions, but in a way that does not encourage further contact. A brief response to students such as "I am here to observe," usually will satisfy them. After a while, students usually become accustomed to the coders' presence and pay little attention to them.

(6) Coders must avoid being drawn into class activities by the teacher or students. They should not help students with their school work or in any way assume a teaching role. If a teacher attempts to include the coder in classroom activities, or put the coder in charge of the class while she leaves the room, the coder should speak to the teacher at the earliest possible moment and explain that this is incompatible with the necessary methods of data collection.

(7) Some teachers will question the coders about the observational system and the goals of the research project. Coders should be briefed on how to respond to such queries. Some teachers ask questions simply out of curiosity; others do so out of anxiety, or the fear that they are being "evaluated." Coders should be able to react to these teachers with non-threatening answers and a straight-forward manner. If the design of the study permits it, the coder can show blank coding sheets to the teacher and very briefly describe the larger sections of the system. However, this would only need to be done with teachers who show an intense interest or a high level of anxiety regarding the data being collected. There is no one "right" way to respond to teacher questions, since each research project will vary in how much information the researchers want the teachers to know, and how much the coders themselves know. In <u>any</u> project, however, coders should be instructed in the <u>specifics</u> of how to answer questions from the teachers being observed.

(8) One of the primary concerns of teachers and principals regarding research projects in their school is that the data collected will be kept strictly confidential. In order that this may be done, coders should keep

in mind several "don'ts": (1) don't talk to any teacher or principal about other teachers or students, (2) don't talk about teachers or their students with other coders, and (3) don't talk about teachers, students, or the research project outside of work-related discussions with project staff.

(9) Coders should take their manuals into the class with them during the training period in case they need to refer to it. If an interaction or event occurs that coders are unsure how to code, they should note it and discuss it later during meetings with the trainer.

During the actual data collection, any irregularities or special problems encountered while coding should also be noted and discussed later with the trainer. If these special situations are found to affect the data quality, the correct ways of handling them should be communicated to all other coders through a memo or during a group meeting.

(10) Coders should always check their coding sheets before turning them in. All identification information, such as school, teacher, class period,etc., should be completed. Also, the data should be scanned for incomplete coding or mistakes. This is of special importance in large scale studies, because no one may be able to remember later who observed what teacher when, and coders may not remember what happened during a particular observation.

(11) Coders should always take along plenty of coding sheets when collecting data. Also, coders should use pencil, <u>not pen</u>, while coding, so that mistakes may be corrected easily. Folding lap boards with large spring clips are ideal for coding since they can hold seating charts and extra coding sheets, and also provide a good flat surface to work on.

Training in the Classroom

When trainees have familiarized themselves thoroughly with the use of the observational system, they are ready to begin in-classroom coding practice. If possible, coders should practice in the same classrooms they will be observing later. This allows the coders to receive more "relevant" experience: they can become accustomed to the teacher's style of teaching and the classroom routine. Coding a particular class is less difficult if the coder already knows what to expect about the pace of events and types of activities. Also, this allows the teacher and students an opportunity to get accustomed to the presence of an observer. It will help to minimize "observer effect" in the succeeding periods of data collection if the presence of a certain coder does not introduce an entirely novel element into the class atmosphere. This is especially important if the teacher is to be observed only a few times since the class will have less opportunity to adjust to the coder's presence.

Practicing in pairs is preferable to coders going into classrooms alone for the first time. During the class, coders may be able to compare their coding as they go along, can point out each others errors, and in general combine their knowledge and skills to help one another make the more difficult discriminations and decisions. Of course, any conversation between the coders must be carried on in a very low tone of voice so that it will not disturb the class.

After the class is over, the pair should thoroughly discuss any problems, questions, or systematic differences in their coding. Referring to the coding manual will solve most difficulties. However, coders always

should note any interaction they were not sure how to code, so that it may be discussed later with the trainer and with other coders. During the first days of in-classroom training, the coders should meet with the trainer every day to discuss their progress and/or problem areas. Hearing questions answered and problems discussed, coders will be able to profit from the experiences of others. This helps to insure that all coders are using the system in the same way.

Trainees should not attempt to code for great lengths of time at first. One or two hours of coding a day followed by opportunities to ask questions, discuss difficulties, etc., will be more effective than coding for 4 or 5 hours. The more complex the observational system, the longer it will take for coders to get to the point where they can code for long periods of time without being overwhelmed with doubts and confusion.

If the individual student is to be the unit of analysis, coders will be recording interactions for specific students. These students (called "target students") will need to be assigned individual identification numbers. For coders to match these students with their numbers, a classroom seating chart will have to be obtained from the teacher. If the students do not sit in assigned seats, the coder will have to make arrangements with the teacher concerning identification of the target students. The coder can then write out short descriptions of each target student. For example, a student may be described this way: #14--Jane Smith--tall, wears glasses, red hair.

In the beginning of data collection, the coding of target students is a little difficult for coders, because they must constantly refer to the seating chart or list of descriptions as they are coding. In fact, during the first few days of practice coding in the classroom, trainees usually

should not attempt to use seating charts or descriptions. This will only slow them down and interfere with learning to code. After the coders have attained good facility with the observational system, they can begin to use seating charts to record target student numbers and interactions with relative ease.

Although coders may be eager to achieve a high degree of inter-coder reliability, the first few days of in-classroom practice should be just that--practice. There is no point in trying to obtain acceptable reliability estimates before coders have begun to build up speed and accuracy and before the inevitable questions and difficulties have been dealt with.

Input From Coders

The project director and/or trainer should be receptive to coder input in the form of complaints and suggestions, especially if the observational system is newly developed or has been adapted from another system. Classroom coders are a valuable source of information--they know what is going on in the classrooms, while the project director has probably not spent time in the classrooms and very likely has never used the observational system. The coders are actually using the system; and what looks good on paper or in theory may be unworkable when employed in a classroom setting. A category may look clear cut and may be important for the goals of the project, but if the coders cannot code it reliably, the data is useless. If coders report that a certain category just doesn't "work," or that additional categories are needed, their suggestions should be seriously considered by the project director.

Reliability Estimates

After coders have spent a few days practicing in the classroom, their data may be used in obtaining inter-coder reliability estimates. Reliability estimates can be expressed as the "percentage of agreement" between two coders. The percentage of agreement is calculated by dividing the number of times coders agreed in their coding of interaction by the sum of these agreements plus the times they disagreed in their coding plus the number of times that one coder coded an interaction and the other did not:

$$\text{percentage agreement} = \frac{\text{Agreements between Coder A and Coder B}}{\text{Agreements for A \& B + disagreements for A \& B + omissions for A + omissions for B}}$$

Reliability should be calculated for each major subsection of the system; then these figures may be averaged to get reliability figures for the observational system as a whole. In our research, we have typically used a criterion of 80% inter-coder agreement. If agreement is good in all but one part of the system, coders will know they are having trouble with this section. The amount of time required to obtain acceptable inter-coder agreement will depend on the complexity of the observational system and the coders' skill in using it.

If some coders reach acceptable standards of reliability in a very short time, they can be paired with coders who are having more difficulty. It will be easier to determine why certain coders are not coding reliably if one member of the pair can be relied upon for the "correct" coding of interactions.

If certain pairs of coders have difficulty reaching reliability, it will be necessary to find out why they disagree in their coding. It may be diffi-

cult to determine the problem. However, there are certain things to be looked for.

Consistent Errors

It is possible that one coder is making a consistent error in coding a certain category. This is the easiest problem to correct once the error has been identified. Coders just need to scan their coding sheets to see if there are certain categories that they coded differently, and determine which is the correct coding by referring to the coding manual and/or by consulting with the trainer or project director.

Over-coding and Under-coding

If reliability is low, check to see if one coder is simply coding more interactions than the other. If this is occurring, it may be that one coder has simply not built up enough speed to keep up with all events in the classroom. More practice will remedy this. The other possibility is that the coder who records more interactions is "over-coding"; that is, recording more interactions than actually occur, or coding inappropriate interactions that the observational system was not designed to pick up. This problem is a little more difficult to correct since it may involve a good deal of misunderstanding on the part of the coder.

Widespread Disagreement

If most pairs of coders have low percentages of agreement there may be several reasons for this:

(1) The coders may not have gained enough skill in using the system. As mentioned earlier, it is pointless to attempt reliability estimates if coders are still having difficulty with their coding. Coders will usually know when they are "ready" for this.

(2) The coding manual, or the observational system itself may be too vague, or, it may not adequately reflect or categorize classroom events. If coders continue to have problems using the system, it might be necessary to (a) "tighten up" the coding manual by adding more specific instructions, (b) establish coding conventions to deal with ambiguous events or interactions, and/or (c) rework the observational system by adding categories (or combining categories if they are difficult for coders to discriminate).

Coder Drift

If data collection continues for several months, the quality of the data may become lower due to "coder drift." Coders working independently may tend to diverge or "drift apart" in their coding. If the possibility of coder drift presents a potential problem, it would be advisable to conduct reliability checks midway through the data collection period. This will not only provide proof that coders are coding as accurately as they previously did, but also, if some coder drift has occurred, provide an opportunity to re-establish reliability and insure that data quality is high.